

Independent claim 1 recites a method causing to commence processing of a second enqueue or dequeue request with respect to a queue prior to completion of processing a prior enqueue or dequeue request with respect to the same queue.

Independent claim 21 recites an article comprising a computer-readable medium that includes computer-readable instructions that cause a computer to commence processing of a received enqueue or dequeue request with respect to a queue prior to completion of processing a prior enqueue or dequeue request with respect to the same queue.

Independent claims 9 and 15 recite, in part, an apparatus or system comprising a queue manager configured to process the enqueue requests and the dequeue requests and capable of commencing processing a request to a queue while a previous request with respect to the same queue is being processed.

As shown in the examples of FIGS. 2, 5 and 6, either an enqueue request 23 or dequeue request 25 is sent through receive ring 22 to a queue manager 27. Each enqueue request 23 represents a request to append a newly received buffer to a queue of buffers 48 (pg. 4, lines 18-20) in a first memory 30. Each dequeue request represents a request to remove a buffer from the queue 48 (pg. 4, lines 29-31). The queue 48 points to data buffers 38 in a second memory. The queue manager 27 processes both enqueue and dequeue requests and manages a cache of queue descriptors 46 which describe the queue 48 (page 5, lines 7-19). Queue descriptors 46 also include pointers 49 for pointing to the queue 48 (page 6, lines 11-14).

In response to receiving an enqueue request 23, the queue 48 is updated to point to a data buffer 38 containing the received data packet. In response to a dequeue request 25, the data buffers 38 currently being pointed to are returned to the queue manager 27 for further processing. In either case, the pointer 49 is updated to point to the next buffer in the queue 48 and to reflect the number of buffers in the queue 48 (page 9, lines 28-30). Once the pointer 49 has been updated, a second pair of dequeue or enqueue requests can be started (page 12, lines 13-20) prior to completion of the first enqueue or dequeue request.

The Office action alleges that the Calvignac patent teaches a method of causing to commence processing of a second enqueue or dequeue request with respect to a queue prior to completion of processing a first enqueue or dequeue request with respect to that queue. Applicant respectfully disagrees.

Calvignac discloses a method for reserving frame modification information in a data storage unit in which a packet processor 100 includes a data flow unit 110 configured to receive frames of data. Each frame of data is associated with a Frame Control Block (FCB) that describes the associated frame of data (col. 3, line 67 – col. 4, lines 1-2). FCBs stored temporarily in flow queues 223 are transmitted to a Target Blade Queue (TBQ) 215 (col. 6, lines 13-17). A FCB queued in the Target Blade Queues 215 subsequently may be scheduled to be dequeued from Target Blade Queue 215 and enqueued into a Port Control Block (PCB) 230 by Target Blade Queue scheduler 228. Once the FCB is enqueued into the Port Control Block 230, the Port Control Block 230 issues a request to a memory arbiter 204 to read data at the beginning of the frame (col. 6, lines 18-26).

The Office action alleges in page 2, section 3 that configuring the Target Blade Queue scheduler 228 to enqueue the FCB into Port Control Block 230 corresponds to the claimed second enqueue request and dequeuing the FCB from Target Blade Queue 215 corresponds to the first enqueue request. Furthermore, the Office action alleges the second request is commenced prior to completion of the first request. This is incorrect.

Claim 1 recites the first and second enqueue and dequeue requests are received in the queue manager with respect to the *same* queue. Calvignac, however, discloses in col. 6, lines 18-22, that the request to dequeue the FCB is with respect to the Target Blade Queue 215 and the subsequent request to enqueue the same FCB is with respect to Port Control Block 230. Target Blade Queue 215 and Port Control Block 230 represent different queues and cannot be considered the *same* queue. Therefore, Calvignac does not disclose or suggest first and second enqueue or dequeue requests are received with respect to the same queue.

Even if the enqueue and dequeue requests were made with respect to the same queue such as, for example, Target Blade Queue 215, these requests are not received by the same queue manager. Instead, Calvignac et al. discloses that FCBs are *enqueued* into Target Blade Queue 215 by scheduler 130 and Target Blade Queue enqueue logic 227 and are *dequeued* from Target Blade Queue 215 by TBQ scheduler 228 (col. 6, lines 13-20). Calvignac et al., therefore, does not disclose or suggest that the FCB enqueue and dequeue requests with respect to Target Blade Queue 215 are received by the same queue manager as recited by claim 1 of the present application.

Furthermore, contrary to the statements in the Office action, there is no disclosure or suggestion of processing the second request prior to completion of the request.

Wilford et al. discloses a linecard architecture for high speed routing of data in a communications device but does not disclose the features missing from the Calvignac et al. patent.

In view of the foregoing remarks, applicant respectfully requests reconsideration and withdrawal of the rejection of independent claims 1, 9, 15 and 21.

Claims 2-8, 10-14, 16-20 and 22-28 depend from claims 1, 9, 15 and 21 and should be allowed for at least the same reasons.

The dependent claims recite additional features that make these claims independently patentable. For example, claim 2 of the present application recites a method of claim 1 that includes causing to modify stored information describing a structure of the queue. Claims 14, 20 and 22 recite similar features. The Office action alleges that the frame data of Calvignac et al. stored in data storage unit 140 corresponds to the claimed stored information describing a structure of the queue. Applicants respectfully disagree. Neither the frame data stored in data storage unit 140 nor the data storage unit 140 describe a structure of any queue. Calvignac et al. discloses that data control unit 140 simply stores the frame data (col. 5, lines 1-2) and the frame

data is passed between senders and receivers in a network (col. 1, lines 27-32). Calvignac et al. does not suggest or disclose the frame data describes a structure of any queue or that the frame data itself is modified.

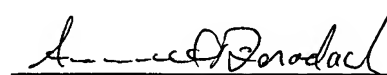
For this additional reason, claim 2 should be allowable.

It is believed all of the pending claims have been addressed. However, the absence of a reply to a specific rejection, issue or comment does not signify agreement with or concession of that rejection, issue or comment. In addition, because the arguments made above may not be exhaustive, there may be reasons for patentability of any or all pending claims (or other claims) that have not been expressed. Finally, nothing in this paper should be construed as an intent to concede any issue with regard to any claim, except as specifically stated in this paper, and the amendment of any claim does not necessarily signify concession of unpatentability of the claim prior to its amendment.

Enclosed is a \$120.00 check for the Petition for Extension of Time fee. Please apply any other charges or credits to deposit account 06-1050.

Respectfully submitted,

Date: 1/30/06



Samuel Borodach
Reg. No. 38,388

Fish & Richardson P.C.
12390 El Camino Real
San Diego, CA 92130
Telephone: (858) 678-5070
Facsimile: (858) 678-5099